CLAIMS

What is claimed is:

1. A device controller for establishing a communication channel between a slave device and a host device, the slave device including the device controller and a function engine, the slave device and host device being connected to a packet-switched serial bus, the device controller comprising:

a serial interface engine having a serial port that connects to the serial bus and a data port, the serial interface engine for generating and interpreting packets on the serial bus and transferring data between the serial bus and the data port; and

an interfacing device connected between the data port of the serial interface engine and the function engine to transfer data between the serial interface engine and the function engine, the interfacing device including a configuration module for configuring the communication channel between the slave device and the host device.

2. A device controller as recited in claim 1, wherein the interfacing device further includes:

at least one register that stores configuration information relating to the communication channel between the slave device and the host device; and

at least one memory that holds operating data relating to the communication channel;

wherein the configuration module is connected to the at least one memory and includes a plurality of finite state machines that are operative to receive and respond to a request from the host device.

3. A device controller as recited in claim 2, wherein the configuration module includes:

a first state machine for receiving and storing a request packet from the host, the first state machine having an output line carrying a signal indicating that a request packet has been stored and is available to be interpreted;

a second state machine for accessing and interpreting the request packet in response to the output signal of the first finite state machine, the second state machine having at least one output line for carrying a data transfer signal, and activating the data transfer signal based on the interpretation of the request packet; and

at least one data transfer state machine for transferring requested data in response to an active data transfer signal.

4. A device controller as recited in claim 3,

wherein the request is a GET_DESCRIPTOR request; and

wherein the data transfer state machine delivers at least a seventeen-byte descriptor to the serial interface engine to be sent to the host device.

5. A device controller as recited in claim 3,

wherein there are a plurality of data transfer state machines; and

wherein only one data transfer state machine receives an active data transfer signal.

6. A device controller as recited in claim 3,

wherein the type of request interpreted by the second state machine is not supported; and

wherein the second state machine indicates to the serial interface engine to issue a stall packet to the host.

7. A device controller as recited in claim 1, further comprising:

at least one endpoint associated with the function engine; and

a group of state machines associated with the function-engine endpoint.

8. A device controller as recited in claim 7,

wherein the endpoint includes an endpoint register having a type field for indicating the type of data transfer the endpoint supports; and

wherein the type field is accessible by at least one of the state machines in the group associated with the endpoint.

9. A device controller as recited in claim 7,

wherein the group of state machines has a clock line for carrying a clock signal;

wherein the serial interface engine detects a start-of-frame condition on the serial bus; and

wherein the clock signal for the group of state machines is derived from the start-of-frame condition.

10. A device controller as recited in claim 7,

wherein the function-engine endpoint is an OUT-type endpoint; and

wherein the group of state machines associated with the function engine endpoint includes:

a data storage state machine that holds data sent from the host for the function engine; and

a command state machine that interprets the stored data in the data storage machine to operate the function engine.

11. A device controller as recited in claim 10, wherein the function engine includes a D/A converter.

12. A device controller as recited in claim 7,

wherein the function-engine endpoint is an IN-type endpoint; and

wherein the group of state machines associated with the function engine endpoint includes a data collecting state machine that holds data sent from the function engine for the host; and

a command state machine that interprets commands from the host to read the data in the data collecting state machine.

13. A device controller as recited in claim 12, wherein the function engine includes an A/D converter.

14. A device controller as recited in claim 7, wherein the endpoint has a register with a bit indicating whether the endpoint is ready to communicate with the host, a not-ready indication instructing the serial interface engine to send an auto-NAK to the host.

15. A device controller as recited in claim 14, wherein at least one of the state machines in the group determines that the endpoint is ready for communication with the host and sets the bit in the register to indicate that the endpoint is ready.

16. A device controller as recited in claim 1, further comprising:

at least two endpoints associated with the function engine; and

a group of state machines associated with each function-engine endpoint.

17. A device controller as recited in claim 16,

wherein the first endpoint associated with the function engine is an OUT-type endpoint; and

wherein the group of state machines associated with the function engine endpoint includes a data transfer state machine which receives commands from the host for controlling the operations of the second endpoint associated with the function engine.

18. A device controller as recited in claim 1, further comprising

at least endpoint associated with the function engine; and

a memory buffer having a plurality of storage locations and associated with the endpoint; and

an endpoint register having an index field for addressing the memory buffer.

19. A device controller as recited in claim 18, further comprising a counter that is updated every time one of the plurality of the storage locations in the memory buffer is accessed.

Add B3